(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 043 665 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
11.10.2000 Bulletin 2000/41

(51) Int. Cl.7: G06F 17/30

(21) Application number: 00106164.7

(22) Date of filing: 21.03.2000

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: 09.04.1999 US 288724

(71) Applicant:
International Business Machines Corporation
Armonk, NY 10504 (US)

(72) Inventors:
- Beigi, Homayoon Sadr Mohammed,
IBM U. K. Ltd.
Winchester, Hampshire SO21 2JN (GB)
- Tritschler, Alain Charles Louis,
IBM U. K. Ltd.
Winchester, Hampshire SO21 2JN (GB)
- Viswanathan, Mahesh,
IBM United Kingdom Ltd.
Winchester, Hampshire SO21 2JN (GB)

(74) Representative:
Zerbi, Guido Maria
Intellectual Property Department,
IBM United Kingdom Ltd.,
Hursley Park
Winchester, Hampshire SO21 2JN (GB)

# (54) Methods and apparatus for retrieving audio information using content and speaker information

(57) Methods and apparatus are disclosed for retrieving audio information based on the audio content as well as the identity of the speaker. The results of content and speaker-based audio information retrieval methods are combined to provide references to audio information (and indirectly to video). A query search system retrieves information responsive to a textual query containing a text string (one or more key words), and the identity of a given speaker. An indexing system transcribes and indexes the audio information to create time-stamped content index file(s) and speaker index file(s). An audio retrieval system uses the generated content and speaker indexes to perform query-document matching based on the audio content and the speaker identity. Documents satisfying the user-specified content and speaker constraints are identified by comparing the start and end times of the document segments in both the content and speaker domains. Documents satisfying the user-specified content and speaker constraints are assigned a combined score that can be used in accordance with the present invention to rank-order the identified documents returned to the user, with the best-matched segments at the top of the list.



Printed by Xerox (UK) Business Services
2.16.7 (HRS)/3.6

## Description

### Field of the Invention

[0001] The present invention relates generally to information retrieval systems and, more particularly, to methods and apparatus for retrieving multimedia information, such as audio and video information, satisfying user-specified criteria from a database of multimedia files.

### Background of the Invention

[0002] Information retrieval systems have focused primarily on retrieving text documents from large collections of text. The basic principles of text retrieval are well established and have been well documented. See, for example, G. Salton, Automatic Text Processing, Addison-Wesley, 1989. An index is a mechanism that matches descriptions of documents with descriptions of queries. The indexing phase describes documents as a list of words or phrases, and the retrieval phase describes the query as a list of words or phrases. A document (or a portion thereof) is retrieved when the document description matches the description of the query.

[0003] Data retrieval models required for multimedia objects, such as audio and video files, are quite different from those required for text documents. There is little consensus on a standard set of features for indexing such multimedia information. One approach for indexing an audio database is to use certain audio cues, such as applause, music or speech. Similarly, an approach for indexing video information is to use key frames, or shot changes. For audio and video information that is predominantly speech, such as audio and video information derived from broadcast sources, the corresponding text may be generated using a speech recognition system and the transcribed text can be used for indexing the associated audio (and video).

[0004] Currently, audio information retrieval systems consist of two components, namely, a speech recognition system to transcribe the audio information into text for indexing, and a text-based information retrieval system. Speech recognition systems are typically guided by three components, namely, a vocabulary, a language model and a set of pronunciations for each word in the vocabulary. A vocabulary is a set of words that is used by the speech recognisor to translate speech to text. As part of the decoding process, the recognisor matches the acoustics from the speech input to words in the vocabulary. Therefore, the vocabulary defines the words that can be transcribed. If a word that is not in the vocabulary is to be recognised, the unrecognised word must first be added to the vocabulary.

[0005] A language model is a domain-specific database of sequences of words in the vocabulary. A set of probabilities of the words occurring in a specific order is also required. The output of the speech recognisor will be biased towards the high probability word sequences when the language model is operative. Thus, correct decoding is a function of whether the user speaks a sequence of words that has a high probability within the language model. Thus, when the user speaks an unusual sequence of words, the decoder performance will degrade. Word recognition is based entirely on its pronunciation, i.e., the phonetic representation of the word. For best accuracy, domain-specific language models must be used. The creation of such a language model requires explicit transcripts of the text along with the audio.

[0006] Text-based information retrieval systems typically work in two phases. The first phase is an off-line indexing phase, where relevant statistics about the textual documents are gathered to build an index. The second phase is an on-line searching and retrieval phase, where the index is used to perform query-document matching followed by the return of relevant documents (and additional information) to the user. During the indexing phase, the text output from the speech recognition system is processed to derive a document description that is used in the retrieval phase for rapid searching.

[0007] During the indexing process, the following operations are generally performed, in sequence: (i) tokenization, (ii) part-of-speech tagging, (iii) morphological analysis, and (iv) stop-word removal using a standard stop-word list. Tokenization detects sentence boundaries. Morphological analysis is a form of linguistic signal processing that decomposes nouns into their roots, along with a tag to indicate the plural form. Likewise, verbs are decomposed into units designating person, tense and mood, along with the root of the verb. For a general discussion of the indexing process, see, for example, S. Dharanipragada et al., "Audio-Indexing for Broadcast News," in Proc. SDR97, 1997.

[0008] While such content-based audio information retrieval systems allow a user to retrieve audio files containing one or more key words specified in a user-defined query, current audio information retrieval systems do not allow a user to selectively retrieve relevant audio files based on the identity of the speaker. Thus, a need exists for a method and apparatus that retrieves audio information based on the audio content as well as the identity of the speaker.

[0009] It is an object of the invention to provide a technique which alleviates the above drawbacks.

## Summary of the Invention

[0010] According to the present invention we provide a method for retrieving audio information from one or more audio sources, said method comprising the steps of:

receiving a user query specifying at least one content and one speaker constraint; and
comparing said user query with a content index and a speaker index of said audio source to identify audio information satisfying said user query.

[0011] Also according to the present invention we provide an audio retrieval system for retrieving audio information from one or more audio sources, comprising: a memory that stores a content index and a speaker index of said audio source and computer-readable code; and a processor operatively coupled to said memory, said processor configured to implement said computer-readable code, said computer-readable code configured to: receive a user query specifying one or more words and the identity of a speaker; and combine the results of a content-based and a speaker-based audio information retrieval to provide references to said audio source based on the audio content and the speaker identity.

[0012] A more complete understanding of the present invention, as well as further features and advantages of the present invention, will be obtained by reference to the following detailed description and drawings.

## Brief Description of the Drawings

[0013]

FIG. 1 is a block diagram of an audio retrieval system according to the present invention;

FIG. 2A is a table from the document database of the content index file(s) of FIG. 1;

FIG. 2B is a table from the document chunk index of the content index file(s) of FIG. 1;

FIG. 2C is a table from the unigram file (term frequency) of the content index file(s) of FIG. 1;

FIG. 2D is a table from the an inverse document index (IDF) of the content index file(s) of FIG. 1;

FIG. 3 is a table from the speaker index file(s) of FIG. 1;

FIG. 4 illustrates a representative speaker enrollment process in accordance with the present invention;

FIG. 5 is a flow chart describing an exemplary indexing system process, performed by the audio retrieval system of FIG. 1; and

FIG. 6 is a flow chart describing an exemplary content and speaker audio retrieval system process, performed by the audio retrieval system of FIG. 1.

## Detailed Description of Preferred Embodiments

[0014] An audio retrieval system 100 according to the present invention is shown in FIG. 1. As discussed further below, the audio retrieval system 100 combines the results of two distinct methods of searching for audio material to provide references to audio information (and indirectly to video) based on the audio content as well as the identity of the speaker. Specifically, the results of a user-specified content-based retrieval, such as the results of a Web search engine, are combined in accordance with the present invention with the results of a speaker-based retrieval.

[0015] The present invention allows a query search system to retrieve information responsive to a textual query containing an additional constraint, namely, the identity of a given speaker. Thus, a user query includes a text string containing one or more key words, and the identity of a given speaker. The present invention compares the constraints of the user-defined query to an indexed audio and/or video database and retrieves relevant audio/video segments containing the specified words spoken by the given speaker.

[0016] As shown in FIG. 1, the audio retrieval system 100 of the present invention consists of two primary components, namely, an indexing system 500 that transcribes and indexes the audio information, and an audio retrieval system 600. As discussed further below, the indexing system 500 processes the text output from a speech recognition

system during the indexing phase to perform content indexing and speaker indexing. During the retrieval phase, the content and speaker audio retrieval system 600 uses the content and speaker indexes generated during the indexing phase to perform query-document matching based on the audio content and speaker identity and to return relevant documents (and possibly additional information) to the user.

[0017] As discussed below, the speech recognition system produces transcripts with time-alignments for each word in the transcript. Unlike a conventional information retrieval scenario, there are no distinct documents in the transcripts and therefore one has to be artificially generated. In the illustrative embodiment, for the content-based index, the transcribed text corresponding to each audio or video file is automatically divided into overlapping segments of a fixed number of words, such as 100 words, and each segment is treated as a separate document. In an alternative implementation, topic identification schemes are used to segment the files into topics. Likewise, for the speaker-based index, the audio or video file is automatically divided into individual segments associated with a given speaker. Thus, a new segment is created each time a new speaker speaks.

[0018] The present invention establishes the best portions of the audio as determined by the content-based retrieval and the speaker-based retrieval. It is noted that the size of a segment in the content based index is about the time it takes to speak 100 words, which is approximately 30 seconds. The length of a segment in the speaker-based index, however, is variable, being a function of the speaker change detector. Thus, the segment length cannot be predicted. Thus, according to a feature of the present invention, the start and end times of the segments in both domains are compared.

[0019] According to a further feature of the present invention, the extent of the overlap between the content and speaker domains is considered. Those documents that overlap more are weighted more heavily. Generally, as discussed further below in conjunction with FIG. 6, the combined score is computed using the following equation:

$$combinedscore=(rankeddocumentscore+(lambda*speakersegmentscore))\ *overlapfactor$$

[0020] The ranked document score ranks the content-based information retrieval, for example, using the Okapi equation, discussed below. The ranked document score is a function of the query terms, and is thus calculated at retrieval time. The speaker segment score is a distance measure indicating the proximity between the speaker segment and the enrolled speaker information and can be calculated during the indexing phase. Lambda is a variable that records the degree of confidence in the speaker identity process, and is a number between zero and one. The overlap factor penalises segments that do not overlap completely, and is a number between zero and one. The combined score can be used to rank-order the identified documents returned to the user, with the best-matched segments at the top of the list.

[0021] FIG. 1 is a block diagram showing the architecture of an illustrative audio retrieval system 100 in accordance with the present invention. The audio retrieval system 100 may be embodied as a general purpose computing system, such as the general purpose computing system shown in FIG. 1. The audio retrieval system 100 includes a processor 110 and related memory, such as a data storage device 120, which may be distributed or local. The processor 110 may be embodied as a single processor, or a number of local or distributed processors operating in parallel. The data storage device 120 and/or a read only memory (ROM) are operable to store one or more instructions, which the processor 110 is operable to retrieve, interpret and execute.

[0022] The data storage device 120 preferably includes an audio corpus database 150 for storing one or more audio or video files (or both) that can be indexed and retrieved in accordance with the present invention. In addition, the data storage device 120 includes one or more content index file(s) 200 and one or more speaker index file(s) 300, discussed below in conjunction with FIGS. 2 and 3, respectively. Generally, as discussed below in conjunction with FIGS. 2A through 2D, the content index file(s) 200 includes a document database 210 (FIG. 2A), a document chunk index 240 (FIG. 2B), a unigram file (term frequency) 260 (FIG. 2C) and an inverse document index (IDF) 275 (FIG. 2D). The content index file(s) 200 are generated in conjunction with a speech recognition system during an indexing phase and describes the audio (or video) documents as a list of words or phrases, together with additional indexing information. The speaker index file(s) 300 is generated in conjunction with a speaker identification system during the indexing phase and provides a speaker label for each segment of an audio file. Thereafter, during the retrieval phase, the content index file(s) 200 and speaker index file(s) 300 are accessed and a document is retrieved if the document description in the content index file(s) 200 matches the description of the user-specified query and the speaker identity indicated by the speaker label in the speaker index file(s) 300 matches the designated speaker identity.

[0023] In addition, the data storage device 120 includes the program code necessary to configure the processor 110 as an indexing system 500, discussed further below in conjunction with FIG. 5, and a content and speaker audio retrieval system 600, discussed further below in conjunction with FIG. 6. As previously indicated, the indexing system 500 analyses one or more audio files in the audio corpus database 150 and produces the corresponding content index file(s) 200 and speaker index file(s) 300. The content and speaker audio retrieval system 600 accesses the content index file(s) 200 and speaker index file(s) 300 in response to a user-specified query to perform query-document match-

ing based on the audio content and speaker identity and to return relevant documents to the user.

## INDEX FILES

[0024] As previously indicated, the audio sample is initially transcribed, for example, using a speech recognition system, to produce a textual version of the audio information. Thereafter, the indexing system 500 analyses the textual version of the audio file(s) to produce the corresponding content index file(s) 200 and speaker index file(s) 300.

[0025] As previously indicated, the content index file(s) 200 includes a document database 210 (FIG. 2A), a document chunk index 240 (FIG. 2B), a unigram file (term frequency) 260 (FIG. 2C) and an inverse document index (IDF) 275 (FIG. 2D). Generally, the content index files 200 store information describing the audio (or video) documents as a list of words or phrases, together with additional indexing information. In the illustrative embodiment, the content index file(s) 200 records, among other things, statistics required by the Okapi equation.

[0026] The document database 210 (FIG. 2A) maintains a plurality of records, such as records 211 through 214, each associated with a different 100 word document chunk in the illustrative embodiment. In one implementation, there is a 50 word overlap between documents. For each document chunk identified in field 220, the document database 210 indicates the start and end time of the chunk in fields 222 and 224, respectively, as well as the document length in field 226. Finally, for each document chunk, the document database 210 provides a pointer to a corresponding document chunk index 240, that indexes the document chunk. Although documents have a fixed length of 100 words in the illustrative embodiment, the length in bytes can vary. As discussed below, the document length (in bytes) is used to normalise the scoring of an information retrieval.

[0027] The document chunk index 240 (FIG. 2B) maintains a plurality of records, such as records 241 through 244, each associated with a different word in the corresponding document chunk. Thus, in the illustrative implementation, there are 100 entries in each document chunk index 240. For each word string (from the document chunk) identified in field 250, the document chunk index 240 indicates the start time of the word in field 255.

[0028] A unigram file (term frequency) 260 (FIG. 2C) is associated with each document, and indicates the number of times each word occurs in the document. The unigram file 260 maintains a plurality of records, such as records 261 through 264, each associated with a different word appearing in the document. For each word string identified in field 265, the unigram file 260 indicates the number of times the word appears in the document in field 270.

[0029] The inverse document index 275 (FIG. 2D) indicates the number of times each word appears in the collection of documents (the audio corpus), and is used to rank the relevance of the current document amongst all documents in which the word occurs. The inverse document index 275 maintains a plurality of records, such as records 276 through 279, each associated with a different word in the vocabulary. For each word identified by the vocabulary identifier in field 280, the inverse document index 275 indicates the word string in field 285, the inverse document frequency (IDF) in field 290 and a list of the documents in which the word appears in field 295. The list of documents in field 295 permits a determination of whether the word appears in any documents without actually searching.

[0030] As previously indicated, the speaker index file(s) 300, shown in FIG. 3, provides a speaker label for each segment of an audio file. The speaker index file(s) 300 maintains a plurality of records, such as records 305 through 312, each associated with a different segment of an audio file. Each segment of speech is associated with a different speaker. For each segment identified in field 325, the speaker index file(s) 300 identifies the corresponding speaker in field 330, and the corresponding audio or video file containing the segment in field 335. In addition, the speaker index file(s) 300 also indicates the start and end time of the segment (as offsets from the start of the file) in fields 340 and 345, respectively. The speaker index file(s) 300 indicates a score (distance measure) in field 350 indicating the proximity between the speaker segment and the enrolled speaker information, as discussed below in conjunction with FIG. 5.

## SPEAKER REGISTRATION PROCESS

[0031] FIG. 4 illustrates a known process used to register or enrol speakers. As shown in FIG. 4, for each registered speaker, the name of the speaker is provided to a speaker enrolment process 410, together with a speaker training file, such as a pulse-code modulated (PCM) file. The speaker enrolment process 410 analyses the speaker training file, and creates an entry for each speaker in a speaker database 420. The process of adding speaker's voice samples to the speaker database 420 is called enrolment. The enrolment process is offline and the audio indexing system assumes such a database exists for all speakers of interest. About a minute's worth of audio is generally required from each speaker from multiple channels and microphones encompassing multiple acoustic conditions. The training data or database of enrolled speakers is stored using a hierarchical structure so that accessing the models is optimised for efficient recognition and retrieval.

## INDEXING PROCESS

[0032] As previously indicated, during the indexing phase, the indexing system 500, shown in FIG. 5, processes the text output from the speech recognition system to perform content indexing and speaker indexing. As shown in FIG. 5, the content indexing and speaker indexing are implemented along two parallel processing branches, with content indexing being performed in steps 510 through 535, and speaker indexing being performed during steps 510 and 550 through 575. It is noted, however, that the content indexing and speaker indexing can be performed sequentially, as would be apparent to a person of ordinary skill in the art.

[0033] As an initial step for both content indexing and speaker indexing, capstral features are extracted from the audio files during step 510, in a known manner. Generally, step 510 changes the domain of the audio files to the frequency domain, reduces the dynamic range and performs an inverse transform to return the signal to the time domain.

### Content - Indexing

[0034] The audio information is then applied to a transcription engine, such as the ViaVoice™ speech recognition system, commercially available from IBM Corporation of Armonk, NY, during step 515 to produce a transcribed file of time-stamped words. Thereafter, the time-stamped words are collected into document chunks of a fixed length, such as 100 words in the illustrative embodiment, during step 520.

[0035] The statistics required for the content index file(s) 200 are extracted from the audio files during step 530. As discussed above, the indexing operations includes: (i) tokenization, (ii) part-of-speech tagging, (iii) morphological analysis, and (iv) stop-word removal using a standard stop-word list. Tokenization detects sentence boundaries. Morphological analysis is a form of linguistic signal processing that decomposes nouns into their roots, along with a tag to indicate the plural form. Likewise, verbs are decomposed into units designating person, tense and mood, along with the root of the verb.

[0036] During step 530, the indexing system 500 obtains the statistics required by the Okapi equation. For each word identified in the audio field, the following information is obtained: the term frequency (number of times the word appears in a given document); the inverse document frequency (IDF) (indicating the number of documents in which the word occurs); the document length (for normalisation) and a set of chain linked pointers to each document containing the word (an inverted index).

[0037] The information obtained during step 530 is stored in a content index file(s) 200 during step 535, or if a content index file(s) 200 already exists, the information is updated.

### Speaker - Indexing

[0038] As discussed further below, the speaker-based information retrieval system consists of two components: (1) an acoustic-change detection system (often referred to as speaker segmentation), and (2) a speaker-independent, language-independent, text-independent speaker recognition system. To automate the speaker identification process, the boundaries (turns) between non-homogeneous speech portions must be detected during step 550. Each homogeneous segment should correspond to the speech of a single speaker. Once delineated, each segment can be classified as having been spoken by a particular speaker (assuming the segment meets the minimum segment length requirement required for speaker recognition system).

[0039] The model-selection criterion used to segment the speech during step 550 of the illustrative embodiment, is the well-known Bayesian Information Criterion (BIC). The input audio stream can be modelled as a Gaussian process on the cepstral space. BIC is a maximum likelihood approach to detect (speaker) turns of a Gaussian process. The problem of model identification is to choose one from among a set of candidate models to describe a given data set. It assumes the frames (10 ms) derived from the input audio signal are independent and result from a single-gaussian process. In order to detect if there is a speech change in a window of N feature vectors after the frame i, $1 \leq i <c$ N, two models are built. The first model represents the entire window by one Gaussian, characterised by its mean and full covariance $\{\mu, \Sigma\}$. The second model represents the first part of the window, up to frame i, with a first Gaussian $\{\mu_1, \Sigma_1\}$, and the second part of the window with another Gaussian $(\mu_2, \Sigma_2\}$. The criterion is then expressed as:

$$\Delta BIC(i) = -R(i) + \lambda P, \text{ where}$$

$$R(i) = \frac{N}{2}\log|\Sigma| - \frac{N_1}{2}\log|\Sigma_1| - \frac{N_2}{2}\log|\Sigma_2|$$

$$P=\frac{1}{2}\left(d+\frac{d(d+1)}{2}\right)\log N$$

and

is the penalty associated to the window, $N_1 = i$ is the number of frames of the first part of the window, and $N_2 = (N\text{-}i)$ is the number of frames of the second part; d is the dimension of the frames. Therefore, P reflects the complexity of the models, as $d + \frac{d(d+1)}{2}$ is the number of parameters used to represent the Gaussians.

**[0040]** $\Delta$BIC <0 implies, taking the penalty into account, that the model splitting the window into two Gaussians is more likely than the model representing the entire window with only a single Gaussian. The BIC therefore behaves like a thresholded-likelihood ratio criterion, where the threshold is not empirically tuned but has a theoretical foundation. This criterion is robust and requires no prior training.

**[0041]** In the illustrative implementation, the BIC algorithm has been implemented to make it fast without impairing the accuracy. The feature vectors used are simply mel- cepstra frames using 24 dimensions. No other processing is done on these vectors. The algorithm works on a window-by-window basis, and in each window, a few frames are tested to check whether they are BIC-prescribed segment boundaries. If no segment boundary is found (positive $\Delta$BIC), then the window size is increased. Otherwise, the old window location is recorded, which also corresponds to the start of a new window (with original size).

**[0042]** A detailed set of steps for a BIC implementation is set forth below. The BIC computations are not performed for each frame of the window for obvious practical reasons. Instead, a frame resolution r is used, which splits the window into M = N/r subsegments. Out of the resulting ( M-1 ) BIC tests, the one that leads to the most negative $\Delta$BIC is selected. If such a negative value exists, the detection window is reset to its minimal size, and a refinement of the point detected is performed, with a better resolution. These refinement steps increase the total number of computations and impact the speed-performance of this algorithm. Hence, these should be tailored to the particular user environment, real-time or offline.

**[0043]** If no negative value is found, the window size is increased from $N_{i-1}$ to $N_i$ frames using the following rule" $N_i = N_{i-1} + \Delta N_i$, with $N_i$ also increasing when no change is found: $N_i - N_{i-1} = 2(N_{i-1}-N_{i-2})$. This speeds up the algorithm in homogeneous segments of the speech signal. In order not to increase the error rate though, the $\Delta$Ni has an upper bound. When the detection window gets too big, the number of BIC computations is further reduced. If more than $M_{MAX}$ subsegments are present, only $M_{MAX}$ - 1) BIC computations will be performed -- skipping the first.

**[0044]** During step 555, the results of step 550 are used to analyse the features produced during step 510 and to generate segment utterances, comprised of chunks of speech by a single speaker. The segment utterances are applied during step 560 to a speaker identification system. For a discussion of a speaker identification system, see, for example, H.S.M. Beigi et al., "IBM Model-Based and Frame-By-Frame Speaker-Recognition," in Proc. of Speaker Recognition and Its Commercial and Forensic Applications, Avignon, France (1998). Generally, the speaker identification system compares the segment utterances to the speaker database 420 (FIG. 4) and finds the "closest" speaker.

**[0045]** The speaker identification system has two different implementations, a model-based approach and a frame-based approach with concomitant merits and demerits. The engine is both text and language independent to facilitate live audio indexing of material such as broadcast news.

**Speaker Identification - - The Model-Based Approach**

**[0046]** To create a set of training models for the population of speakers in the database, a model $M_i$ for the $i^{th}$ speaker based on a sequence of M frames of speech, with the d-dimensional

feature vector $\left\{\vec{f}_m\right\}_{m=1,\ldots,M}$, is computed. These models are stored in

terms of their $\left\{\vec{\mu}_{i,j}, \Sigma_{i,j}, \vec{p}_{i,j}\right\}_{j=1,\ldots,n_i}$,

statistical parameters, such as, consisting of the Mean vector, the Covariance matrix, and the Counts, for the case when

a Gaussian distribution is selected. Each speaker, i, may end up with a model consisting of $n_i$ distributions.

**[0047]** Using the distance measure proposed in H.S.M. Beigi et. al, A Distance Measure Between Collections of Distributions and Its Application to Speaker Recognition,' Proc. ICASSP98}, Seattle, WA, 1998, for comparing two such models, a hierarchical structure is created to devise a speaker recognition system with many different capabilities including speaker identification (attest a claim), speaker classification (assigning a speaker), speaker verification (second pass to confirm classification by comparing label with a "cohort" set of speakers whose characteristics match those of the labelled speaker), and speaker clustering.

**[0048]** The distance measure devised for speaker recognition permits computation of an acceptable distance between two models with a different number of distributions $n_i$. Comparing two speakers solely based on the parametric representation of their models obviates the need to carry the features around making the task of comparing two speakers much less computationally intensive. A short-coming of this distance measure for the recognition stage, however, is that the entire speech segment has to be used to build the model of the test individual (claimant) before computation of the comparison can begin. The frame-by-frame approach alleviates this problem.

## Speaker Identification - - The Frame-By-Frame Approach

**[0049]** Let Mi be the model corresponding to the $i^{th}$ enrolled speaker. $M_i$ is entirely defined by the parameter set,

$$\{\vec{\mu}_{i,j}, \Sigma_{i,j}, \vec{p}_{i,j}\}_{j=1,\dots,n_i},$$

consisting of the mean vector, covariance matrix, and mixture weight for each of the $n_i$ components of speaker i's Gaussian Mixture Model (GMM) . These models are created using training data consisting of a sequence of M frames of speech, with the d-dimensional feature vector,

$$\{\vec{f}_m\}_{m=1,\dots,M},$$

as described in the previous section. If the size of the speaker population is $N_p$, then the set of the model universe is $\{M_i\}_{i=1,\dots,N_p}$. The fundamental goal is to find the i such that $M_i$ best explains the test data, represented as a sequence of N frames,

$$\{\vec{f}_n\}_{n=1,\dots,N}$$

, or to make a decision that none of the models describes the data adequately. The following frame-based weighted likelihood distance measure, $d_{i,n}$, is used in making the decision:

$$d_{i,n} = -\log\left[\sum_{j=1}^{n_i} p_{i,j} p\left(\vec{f}_n \middle| j^{th}\ component\ of\ M_i\right)\right]$$

, where, using a Normal representation,

$$p(\vec{f}_n|\cdot) = \frac{1}{(2\pi)^{d/2}|\Sigma_{i,j}|^{1/2}} e^{-\frac{1}{2}(\vec{f}_n - \vec{\mu}_{i,j})^t \Sigma_{i,j}^{-1}(\vec{f}_n - \vec{\mu}_{i,j})}$$

The total distance, $D_i$, of model $M_i$ from the test data is then taken to be the sum of all the distances over the total number of test frames.

**[0050]** For classification, the model with the smallest distance to that of the speech segment is chosen. By comparing the smallest distance to that of a background model, one could provide a method to indicate that none of the original models match very well. Alternatively, a voting technique may be used for computing the total distance.

**[0051]** For verification, a predetermined set of members that form the cohort of the labeled speaker is augmented with a variety of background models. Using this set as the model universe, the test data is verified by testing if the claimant's model has the smallest distance; otherwise, it is rejected.

**[0052]** This distance measure is not used in training since the frames of speech would have to retained for comput-

ing the distances between the speakers. The training is done, therefore, using the method for the model-based technique discussed above.

[0053] The index file for speaker-based retrieval is built by taking a second pass over the results of speaker classification and verification during step 565. If the speaker identification is verified during step 565, then the speaker label is assigned to the segment during step 570. As previously indicated, each classification result is accompanied by a score indicating the distance from the original enrolled speaker model to the audio test segment, the start and end times of the segment relative to the beginning of the audio clip concerned, and a label (name of the speaker supplied during enrolment). In addition, for any given audio clip, all the segments assigned to the same (speaker) label are gathered. They are then sorted by their scores and normalised by the segment with the best score. For every new audio clip processed by the system and added to the index, all the labelled segments are again sorted and re-normalized. This information is stored in a speaker index file(s) 300 during step 575, or if a speaker index file(s) 300 already exists, the information is updated.

## RETRIEVAL PROCESS

[0054] As previously indicated, during the retrieval phase, the content and speaker audio retrieval system 600, shown in FIG. 6, uses the content and speaker indexes generated during the indexing phase to perform query-document matching based on the audio content and speaker identity and to return relevant documents (and possibly additional information) to the user. Generally, retrieval can be performed using two distinct, non-overlapping modules, one for content-based and the other for speaker-based retrieval. The two modules can be programmed to run concurrently using threads or processes since they are completely independent. In the illustrative implementation both modules run sequentially.

[0055] At retrieval time, the content and speaker audio retrieval system 600 loads the same vocabularies, tag dictionaries, morphological tables and token tables that were used in indexing during steps 610 and 20. The appropriate content index file(s) 200 and speaker index file(s) 300 are loaded into memory during step 620. A test is performed during step 625 until a query is received.

[0056] The query string is received and processed during step 630. In response to a received textual query, the query string is compared during step 635 against the content index file(s) 200 to compute the most relevant document(s) using an objective ranking function (ranked document score). The ranked document score that is used in the ranking of these documents is also recorded for subsequent computing of the combined scores in accordance with the present invention (step 645) .

[0057] The following version of the Okapi formula, for computing the ranked document score between a document d and a query q is used:

$$S(d,q)=\sum_{K=1}^{Q} c_q(q_k)\frac{c_d(q_k)}{\alpha_1+\alpha_2\frac{l_d}{l}+c_d(q_k)}\,idf(q_k)$$

Here, $q_k$ is the $k^{th}$ term in the query, Q is the number of terms in the query, $c_q(q_k)$ and $c_d(q_k)$ are the counts of the $k^{th}$ term in the query and document respectively, $l_d$ is the length of the document, l is the average length of the documents in the collection, and $idf(q_kk)$ is the inverse document frequency for the term $q_k$ which is given by:

$$idf(q_k)=\log\left(\frac{N-n(g_k)+0.5}{n(q_k)+0.5}\right),$$

where N is the total number of documents and $n(q_k)$ is the number of documents that contain the term $q_k$. The inverse document frequency term thus favours terms that are rare among documents. (For unigrams, $a_1 = 0.5$ and $a_2 = 1.5$). Clearly, the idf can be pre-calculated and stored as can most of the elements of the scoring function above except for the items relating to the query.

[0058] Each query is matched against all the documents in the collection and the documents are ranked according to the computed score from the Okapi formula indicated above. The ranked document score takes into account the number of times each query term occurs in the document normalised with respect to the length of the document. This normalisation removes bias that generally favour longer documents since longer documents are more likely to have more instances of any given word. This function also favours terms that are specific to a document and rare across other documents. (If a second pass is used, the documents would be re-ranked by training another model for documents, using the top- ranked documents from the first pass as training data.)

[0059] Thereafter, the identified documents (or a subset thereof) are analysed during step 640 to determine if the speaker identified in the speaker index file(s) 300 matches the speaker specified by the user in the query. Specifically, the time bounds of the ranked documents satisfying the content-based query are compared with those documents satisfying the speaker-based query to identify documents with overlapping start and end times. A single segment from speaker retrieval may overlap with multiple segments from text retrieval.

[0060] The combined score for any overlapping documents is computed during step 645 as follows:

Combinedscore=(rankeddocumentscore+(lambda*speakersegmentscore))

in the manner described above. All of the scored documents are then ranked and normalised with the most relevant document getting a match-score of 100.

[0061] Generally, the top N documents alone are returned to the user. Thus, a list of start and end times of the N best-matched segments, together with the match-scores, and the matched words that contributed to the relevance score are returned during step 650. The default start time of each combined result is the same as the start time for the corresponding document from the content-based search. (The other choice is to use the start time of the speaker segment.) The end time is set to the end of the speaker segment (simply to let the speaker finish his statement). However, for usability reasons, the segment can be truncated at a fixed duration, such as 60 seconds, i.e., two times as long as the average document length.

## USER INTERFACE

[0062] The illustrative user interface is capable of showing all the relevant information for each of the N selections returned by the retrieval engine, and on further selection uses a media handler component, implemented using the Java Media Filter, to display MPEG-1 video via a VCR-like interface. The Java application is responsible for locating the video files (which can be on a server if the PC is networked), and then uses information gathered during retrieval to embellish the results, such as displaying the retrieved document, associated information such as media file name, start time, end time, rank, normalised score, a graphic view of where in the media file the retrieved segment lies, highlighting the query words (and other morphs that contributed to the ranking of that document) - this is relevant only for content-based searching, or permitting highlighting of portion of the displayed retrieved document for play back.

[0063] The top N retrieved items are presented to the user in a compact form. This lets the user visually review the retrieved item for further action. Generally, it includes all the gathered information about the retrieved document including a portion of the text of the document. When one of the retrieved items is selected for perusal of the audio or video, the media handler component is called upon to locate the media file, advance to the specified start time, decompress the stream (if required), and then initialise the media player with the first frame of the audio or video. The VCR-like interface permits the user to "play" the retrieved video from start to finish or to stop and advance at any juncture.

[0064] Further improvements can be made within the context of our approach to content-based information retrieval from audio. The current set of documents derived from the speech recognition output can be augmented by including the next-best guesses for each word or phrase from the recognisor. This information can be used for weighting the index terms, query expansion, and retrieval. Also, better recognition accuracy can be had by detecting segments with music or mostly noise that only pure speech is indexed for retrieval. One limitation with the current approach to audio-indexing is the finite coverage of the vocabulary used in the speech recognisor. Words such as proper nouns and abbreviations that are important from an information retrieval standpoint are often found missing in the vocabulary and hence in the recognised transcripts. One method to overcome this limitation is to complement the speech recognisor with a wordspotter for the out of vocabulary words. For this approach to be practical, however, one has to have the ability to detect spoken words in large amounts of speech at speeds many times faster than real-time.

## Claims

1. A method for retrieving audio information from one or more audio sources, said method comprising the steps of:

   receiving a user query specifying at least one content and one speaker constraint; and
   comparing said user query with a content index and a speaker index of said audio source to identify audio information satisfying said user query.

2. The method of claim 1, wherein said content index and said speaker index are time-stamped and said comparing step further comprises the step of comparing the start and end times of the document segments in both the content and speaker domains.

3. The method of any preceding claim, wherein said content index includes the frequency of each word in said audio source.

4. The method of any preceding claim, wherein said content index includes the inverse document frequency (IDF) of each word in said audio source.

5. The method of any preceding claim, wherein said content index includes the length of said audio source.

6. The method of any preceding claim, wherein said content index includes a set of chain linked pointers to each document containing a given word.

7. The method of any preceding claim, wherein said speaker index includes a score indicating the distance from an enrolled speaker model to the audio test segment.

8. The method of any preceding claim, wherein said speaker index includes the start and end times of each audio segment.

9. The method of any preceding claim, wherein said speaker index includes a label identifying the speaker associated with the segment.

10. The method of any preceding claim, wherein said comparing step further comprises the step of comparing documents satisfying the content-based query with documents satisfying the speaker-based query to identify relevant documents.

11. The method of any preceding claim, further comprising the step of transcribing and indexing said audio source to create said content index and said speaker index.

12. The method of claim 11, wherein said step of creating said speaker index comprises the steps of automatically detecting turns in said audio source and assigning a speaker label to each of said turns.

13. The method of any preceding claim, further comprising the step of returning at least a portion of said identified audio information to a user.

14. The method of any preceding claim, further comprising the step of assigning a combined score to each segment of said identified audio information and returning at least a portion of said identified audio information in a ranked-list.

15. The method of claim 14, wherein said combined score evaluates the extent of the overlap between the content and speaker domains.

16. The method of claim 14, wherein said combined score evaluates a ranked document score ranking the content-based information retrieval.

17. The method of claim 14, wherein said combined score evaluates a speaker segment score measuring the proximity between a speaker segment and enrolled speaker information.

18. The method of any preceding claim, wherein said speaker constraint includes the identity of a speaker.

19. The method of any preceding claim, wherein said content constraint includes one or more keywords.

20. An audio retrieval system for retrieving audio information from one or more audio sources, comprising:

a memory that stores a content index and a speaker index of said audio source and computer-readable code; and
a processor operatively coupled to said memory, said processor configured to implement said computer-readable code, said computer-readable code configured to:
receive a user query specifying one or more words and the identity of a speaker; and
combine the results of a content-based and a speaker-based audio information retrieval to provide references to said audio source based on the audio content and the speaker identity.

21. The audio retrieval system of claim 20, wherein said content index and said speaker index are time-stamped and said processor is further configured to compare the start and end times of the document segments in both the content and speaker domains.

22. The audio retrieval system of claim 20 or 21, wherein said content index includes the frequency of each word in said audio source.

23. The audio retrieval system of any claim 20 to 22, wherein said content index includes the inverse document frequency (IDF) of each word in said audio source.

24. The audio retrieval system of any claim 20 to 23, wherein said speaker index includes a score indicating the distance from an enrolled speaker model to the audio test segment.

25. The audio retrieval system of any claim 20 to 24, wherein said speaker index includes a label identifying the speaker associated with the segment.

26. The audio retrieval system of any claim 20 to 25, wherein said processor is further configured to compare documents satisfying the content-based query with documents satisfying the speaker-based query to identify relevant documents.

27. The audio retrieval system of any claim 20 to 26, wherein said processor is further configured to transcribe and index said audio source to create said content index and said speaker index.

28. The audio retrieval system of any claim 20 to 27, wherein said processor is further configured to assign a combined score to each segment of said identified audio information and return at least a portion of said identified audio information in a ranked-list.

29. The audio retrieval system of claim 29, wherein said combined score evaluates the extent of the overlap between the content and speaker domains.

30. The audio retrieval system of claim 29, wherein said combined score evaluates a ranked document score ranking the content-based information retrieval.

31. The audio retrieval system of claim 29, wherein said combined score evaluates a speaker segment score measuring the proximity between a speaker segment and enrolled speaker information.

32. A computer program comprising computer program code means adapted to perform all the steps of any claim 1-19 when said program is run on a computer.

33. The computer program of claim 32 embodied on a computer readable medium.

*FIG. 1*

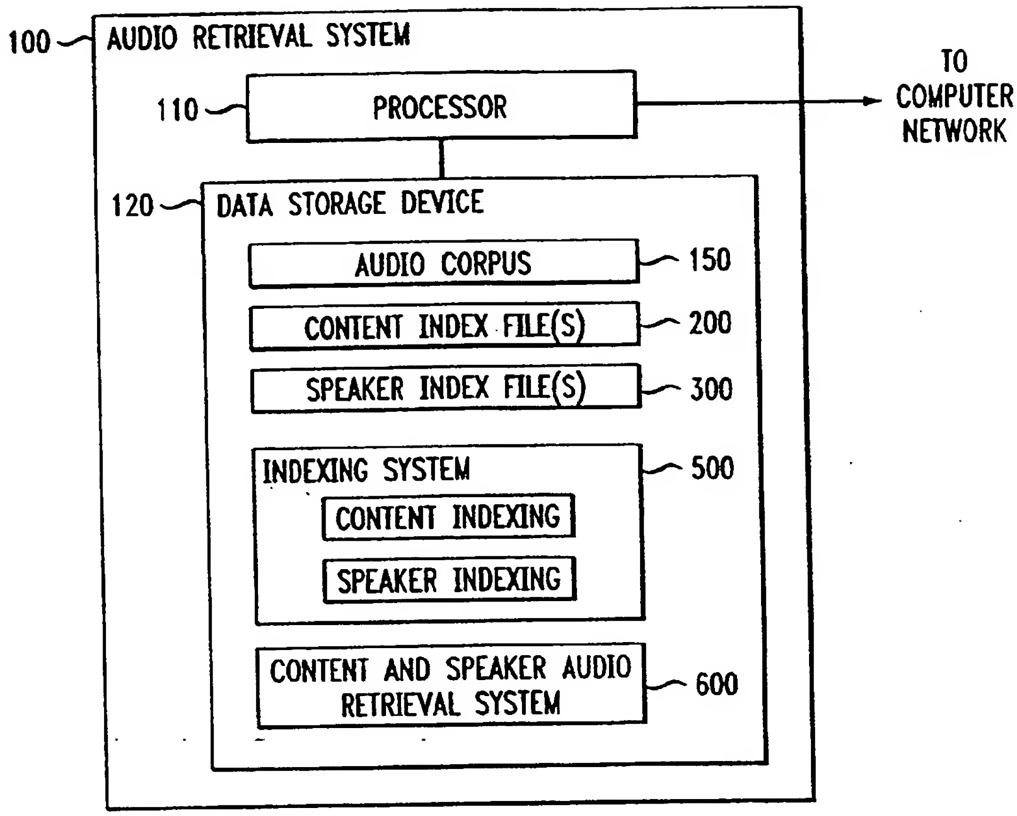
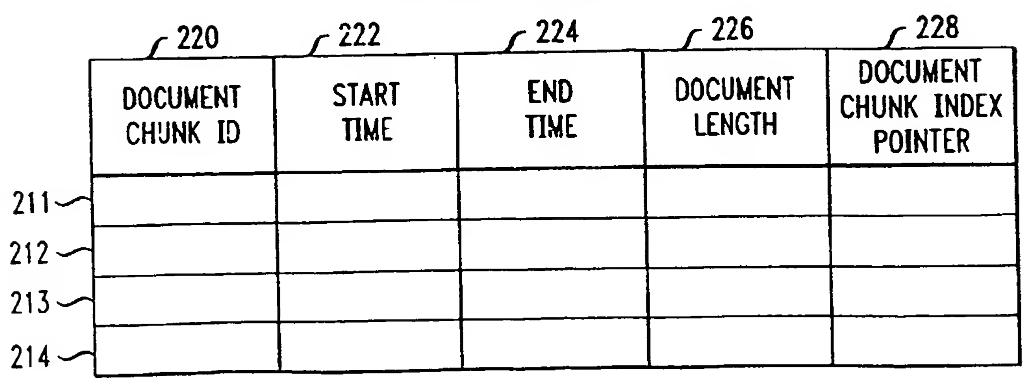



*FIG. 2A*

DOCUMENT DATABASE 210



## FIG. 2B

DOCUMENT CHUNK INDEX
(DOCUMENT CHUNK N1) 240

| 250 | WORD STRING | START TIME | 255 |
|---|---|---|---|
| 241 | 1 | $t_1$ | |
| 242 | 2 | $t_2$ | |
| 243 | ... | ... | |
| 244 | N | $t_N$ | |

## FIG. 2C

UNIGRAM FILE
(TERM FREQUENCY) 260

| 265 | WORD STRING | NUMBER OF OCCURRENCES IN DOCUMENT | 270 |
|---|---|---|---|
| 261 | 1 | $t_1$ | |
| 262 | 2 | $t_2$ | |
| 263 | ... | ... | |
| 264 | N | $t_N$ | |

## FIG. 2D

INVERSE DOCUMENT INDEX 275

| | 280 | 285 | 290 | 295 |
|---|---|---|---|---|
| | VOCABULARY ID | WORD STRING | IDF | DOCUMENT LIST |
| 276 | | | | |
| 277 | | | | |
| 278 | | | | |
| 279 | | | | |

*FIG. 3*

SPEAKER INDEX FILE(S) 300

| | 325 SEGMENT NUMBER | 330 SPEAKER LABEL | 335 AUDIO IDENTIFIER | 340 START TIME | 345 END TIME | 350 SCORE |
|---|---|---|---|---|---|---|
| 305 | 1 | SPEAKER 1 | MEDIA 1 | $T_A$ | $T_B$ | $S_{10}$ |
| 306 | 2 | SPEAKER 1 | MEDIA 6 | $T_K$ | $T_L$ | $S_{11}$ |
| 307 | ... | ... | ... | ... | ... | ... |
| 308 | N | SPEAKER 1 | MEDIA 3 | $T_E$ | $T_F$ | $S_{12}$ |
| | | | | | | |
| 309 | 1 | SPEAKER N | MEDIA 4 | $T_G$ | $T_H$ | $S_{20}$ |
| 310 | 2 | SPEAKER N | MEDIA 5 | $T_I$ | $T_J$ | $S_{21}$ |
| 311 | ... | ... | ... | ... | ... | ... |
| 312 | N | SPEAKER N | MEDIA 7 | $T_M$ | $T_N$ | $S_{22}$ |

*FIG. 4*

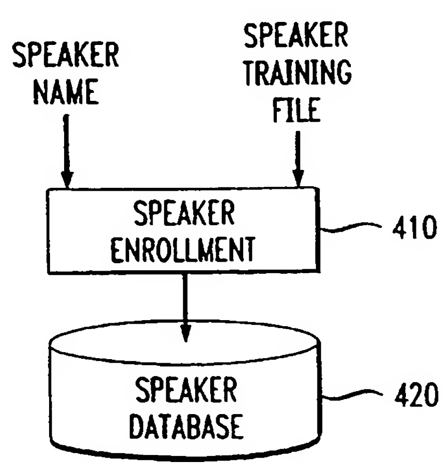


FIG. 5
500

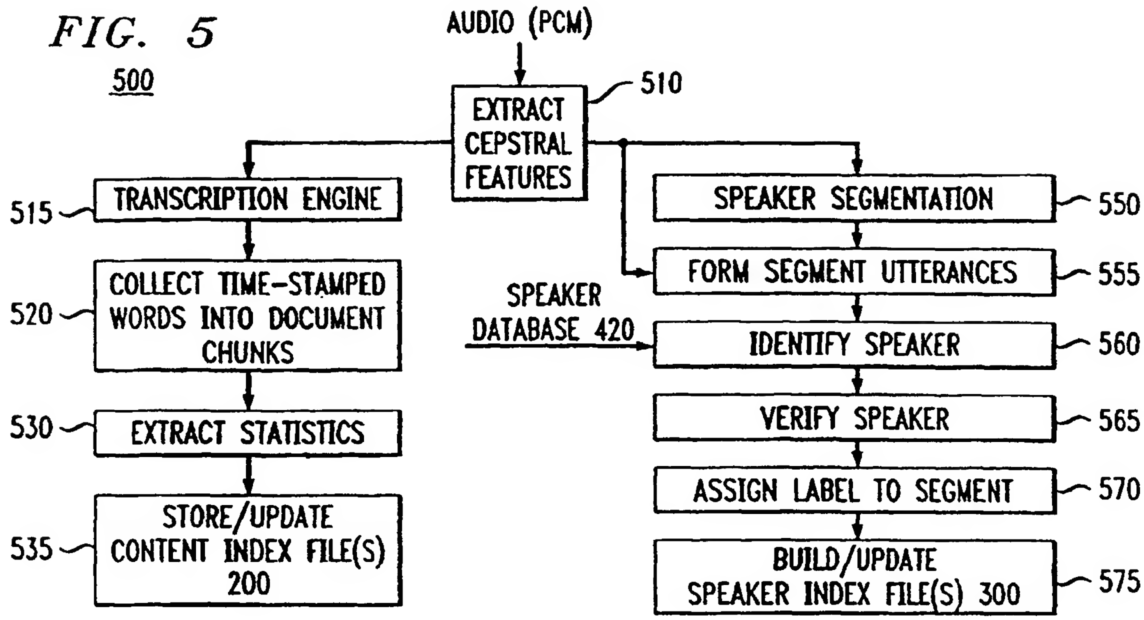
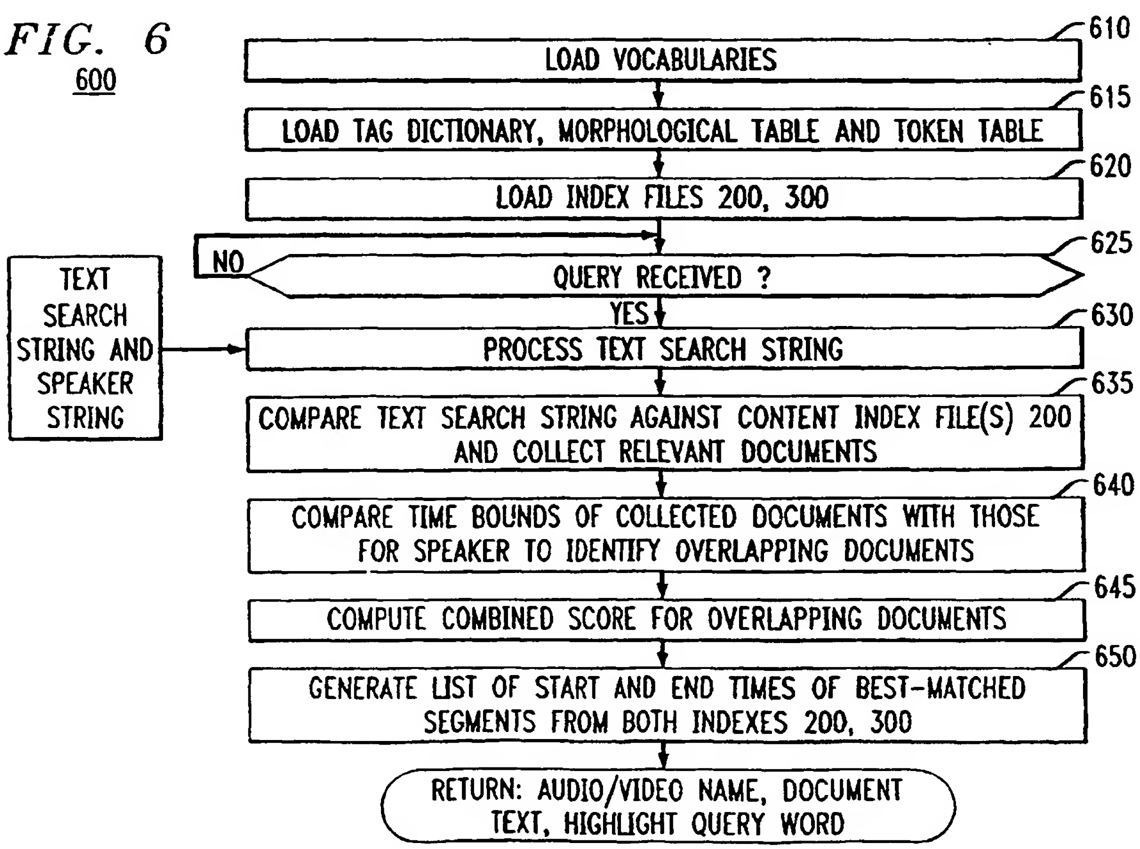



FIG. 6
600



(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 043 665 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
12.02.2003 Bulletin 2003/07

(43) Date of publication A2:
11.10.2000 Bulletin 2000/41

(21) Application number: 00106164.7

(22) Date of filing: 21.03.2000

(51) Int Cl.7: G06F 17/30

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: 09.04.1999 US 288724

(71) Applicant: International Business Machines Corporation
Armonk, NY 10504 (US)

(72) Inventors:
- Beigi, Homayoon Sadr Mohammed, IBM U. K. Ltd.
Winchester, Hampshire SO21 2JN (GB)
- Tritschler, Alain Charles Louis, IBM U. K. Ltd.
Winchester, Hampshire SO21 2JN (GB)
- Viswanathan, Mahesh, IBM United Kingdom Ltd.
Winchester, Hampshire SO21 2JN (GB)

(74) Representative: Zerbi, Guido Maria
Intellectual Property Department,
IBM United Kingdom Ltd.,
Hursley Park
Winchester, Hampshire SO21 2JN (GB)

# (54) Methods and apparatus for retrieving audio information using content and speaker information

(57) Methods and apparatus are disclosed for retrieving audio information based on the audio content as well as the identity of the speaker. The results of content and speaker-based audio information retrieval methods are combined to provide references to audio information (and indirectly to video). A query search system retrieves information responsive to a textual query containing a text string (one or more key words), and the identity of a given speaker. An indexing system transcribes and indexes the audio information to create time-stamped content index file(s) and speaker index file(s). An audio retrieval system uses the generated content and speaker indexes to perform query-document matching based on the audio content and the speaker identity. Documents satisfying the user-specified content and speaker constraints are identified by comparing the start and end times of the document segments in both the content and speaker domains. Documents satisfying the user-specified content and speaker constraints are assigned a combined score that can be used in accordance with the present invention to rank-order the identified documents returned to the user, with the best-matched segments at the top of the list.

EP 1 043 665 A3

European Patent Office

# EUROPEAN SEARCH REPORT

Application Number
EP 00 10 6164

## DOCUMENTS CONSIDERED TO BE RELEVANT

| Category | Citation of document with indication, where appropriate, of relevant passages | Relevant to claim | CLASSIFICATION OF THE APPLICATION (Int.Cl.7) |
|---|---|---|---|
| X | ROY D ET AL: "Speaker identification based text to audio alignment for an audio retrieval system" ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 1997. ICASSP-97., 1997 IEEE INTERNATIONAL CONFERENCE ON MUNICH, GERMANY 21-24 APRIL 1997, LOS ALAMITOS, CA, USA,IEEE COMPUT. SOC, US, 21 April 1997 (1997-04-21), pages 1099-1102, XP010225990 ISBN: 0-8186-7919-0 | 1,2,6, 8-11, 13-16, 18-21, 25-30, 32,33 | G06F17/30 |
| Y | * page 1102, left-hand column, line 1 - line 23 * | 3-5,7, 12,17, 22-24,31 | |
| | * page 1100, left-hand column, line 34 - right-hand column, line 5 * --- | | |
| X | GELIN P ET AL: "Keyword spotting enhancement for video soundtrack indexing" SPOKEN LANGUAGE, 1996. ICSLP 96. PROCEEDINGS., FOURTH INTERNATIONAL CONFERENCE ON PHILADELPHIA, PA, USA 3-6 OCT. 1996, NEW YORK, NY, USA,IEEE, US, 3 October 1996 (1996-10-03), pages 586-589, XP010237864 ISBN: 0-7803-3555-4 * page 586, left-hand column, line 16 - line 26 * --- | 1,20,32, 33 | TECHNICAL FIELDS SEARCHED (Int.Cl.7) G06F G10L |
| Y | ABDEL-MOTTALEB M ET AL: "Aspects of multimedia retrieval" PHILIPS JOURNAL OF RESEARCH, ELSEVIER, AMSTERDAM, NL, vol. 50, no. 1, 1996, pages 227-251, XP004008214 ISSN: 0165-5817 * page 234, line 1-6 * * page 235, line 10-35 * --- -/-- | 3-5,22, 23 | |

The present search report has been drawn up for all claims

| Place of search | Date of completion of the search | Examiner |
|---|---|---|
| BERLIN | 20 November 2002 | Deane, E |

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

# EUROPEAN SEARCH REPORT

Application Number
EP 00 10 6164

## DOCUMENTS CONSIDERED TO BE RELEVANT

| Category | Citation of document with indication, where appropriate, of relevant passages | Relevant to claim | CLASSIFICATION OF THE APPLICATION (Int.Cl.7) |
|---|---|---|---|
| Y | ROY, D.K.: "Speaker Indexing Using Neural Network Clustering of Vowel Spectra" INT. JOURNAL OF SPEECH TECHNOLOGY, [Online] vol. 1, no. 2, March 1997 (1997-03), pages 143-149, XP002221377 Netherlands Retrieved from the Internet: <URL:http://citeseer.nj.nec.com/cache/papers/cs/6196/http:zSzzSzvismod.www.media.mit.eduzSz~dkroyzSzpaperszSzPostscriptzSzijst96.pdf/deb97speaker.pdf> [retrieved on 2002-11-18] * page 1, line 24 - page 2, line 5; figure 1 * --- | 12 | |
| Y | DATABASE INSPEC [Online] THE INSTITUTION OF ELECTRICAL ENGINEERS, STEVENAGE, GB; October 1977 (1977-10) HOLLIEN ET AL.: "Speaker identification by long-term spectra under normal and distorted speech conditions" Database accession no. 1143932 XP002221620 * abstract * --- | 7,17,24,31 | |
| | -/-- | | TECHNICAL FIELDS SEARCHED (Int.Cl.7) |

The present search report has been drawn up for all claims

| Place of search | Date of completion of the search | Examiner |
|---|---|---|
| BERLIN | 20 November 2002 | Deane, E |

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

# EUROPEAN SEARCH REPORT

Application Number
EP 00 10 6164

## DOCUMENTS CONSIDERED TO BE RELEVANT

| Category | Citation of document with indication, where appropriate, of relevant passages | Relevant to claim | CLASSIFICATION OF THE APPLICATION (Int.Cl.7) |
|---|---|---|---|
| A | FOOTE, J.: "An overview of audio information retrieval" MULTIMEDIA SYSTEMS, [Online] vol. 7, no. 1, January 1999 (1999-01), pages 2-10, XP002221393 Germany Retrieved from the Internet: <URL:http://citeseer.nj.nec.com/cache/papers/cs/11284/http:zSzzSzwww.cs.princeton.eduzSzcourseszSzarchivezSzspr99zSzcs598bzSzfoote_over.pdf/foote98overview.pdf> [retrieved on 2002-11-19] * page 7, line 22 - page 8, line 28 * | 1,20,32 | |

TECHNICAL FIELDS SEARCHED (Int.Cl.7)

The present search report has been drawn up for all claims

| Place of search | Date of completion of the search | Examiner |
|---|---|---|
| BERLIN | 20 November 2002 | Deane, E |

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)